# Membership Statistics

Contents

[Membership Statistics 1](#_Toc396737403)

[Column from 4](#_Toc396737404)

[SQL of Count EACH individual member 6](#_Toc396737405)

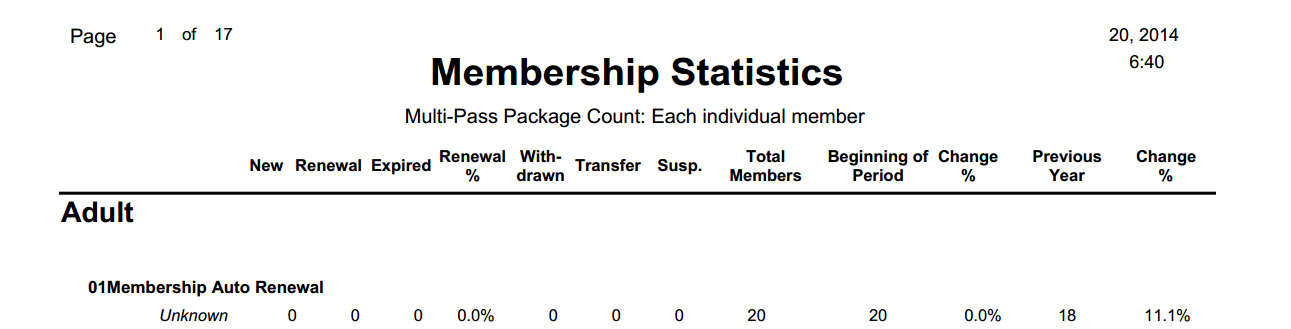
[SQL of Count entire membership as one single unit (1) 11](#_Toc396737406)

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Author** | **Description** |
| August 25, 2014 | 1 | Chandler Xue | Initial |

**Introduction**

To show the statistics information about membership.

## Column from



**New**: The membership’s TRANSACTIONTYPE = 29 to show New

**Renewal**: The membership’s TRANSACTIONTYPE = 33 to show Renewal

**Expired**: The expiration date between start time and end time.

**Renewal%**:Renewal/(Renewal + Expired)

**Withdrawn**: The membership’s TRANSACTIONTYPE = 30 to show Withdrawn

**Transfer**: The membership’s TRANSACTIONTYPE = 31 to show Transfer in and TRANSACTIONTYPE=32 to show Transfer out

**Susp**: The date of Suspended from between start time and end time.

**Total Members**: New – Expired + Transfer

**Beginning Period**: The membership’s effective date less than the start time.

**Change**: (Total Members - Beginning Period)/ Beginning Period

**Previous Year**: The membership’s effective date less than last year’s end time.

**Change**: (Total Members - Previous Year)/ Previous Year

**Related Database Tables**

**PACKAGES** : Record basic information of package, such as package name, category…

Package\_ID is primary key

PackageCategory\_ID is foreign key of relate to PACKAGE\_CATEGORIES

Site\_ID is foreign key of relate to SITES

**PACKAGE\_CATEGORIES :** Record basic information of package category, such as package category name, category description…

PackageCategory\_ID is primary key

**SITES :** Record basic information of site, such as site name, Address…

Site\_ID is primary key

Geographic\_Area\_ID is foreign key of relate to GEOGRAPHIC\_AREAS

POSLayoutID is foregin key of relate to POSLAYOUTPROPERTIES

**CUSTOMERS :** Record basic information of customer, such as customer name, Address…

Customer\_ID is primary key

CustomerType\_ID is foreign key of relate to CUSTOMERTYPES

Geographic\_Area\_ID is foreign key of relate to GEOGRAPHIC\_AREAS

LastPayee\_ID is foreign key of relate to ?

Site\_ID is foreign key of relate to SITES

**CUSTOMER\_MEMBERSHIP\_DATES :** Record the relation between customer and the membership who enroll.

Customer\_Membership\_Date\_ID is primary key

**AGE\_CATEGORIES :** Record basic information of age restriction, such as lower age, upper age…

Age\_Category\_ID is primary key

**MEMBERSHIPS :** Record basic information of membership

Membership\_ID is primary key

Package\_ID is the foreign key of relate to PACKAGES

PrimaryMemberCustomer\_ID is the foreign key of relate to CUSTOMERS

## SQL of Count EACH individual member

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

COUNT(\*) AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

WHERE CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE <= {ts '2014-07-31 00:00:00'}

GROUP BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender)

UNION

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

COUNT(\*) AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

WHERE CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE <{ts '2014-07-01 00:00:00'}

GROUP BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender)

UNION

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

COUNT(\*) AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

WHERE EFFECTIVE\_DATE <{ts '2013-07-31 00:00:00'}

GROUP BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender)

UNION ALL

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

CASE

WHEN EXPIRATION\_DATE >={ts '2014-07-01 00:00:00'}

AND EXPIRATION\_DATE <={ts '2014-07-31 00:00:00'} THEN 1

ELSE 0

END AS EXPIRE,

CASE

WHEN DATESUSPENDEDFROM >={ts '2014-07-01 00:00:00'}

AND DATESUSPENDEDFROM <={ts '2014-07-31 00:00:00'} THEN 1

ELSE 0

END AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID)

UNION ALL (

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1, SITES.sitename AS group2, PACKAGES.packagename AS group3, CUSTOMERS.gender AS group4, CASE WHEN TRANSACTIONTYPE = 29 THEN 1 ELSE 0 END AS NEW, CASE WHEN TRANSACTIONTYPE = 31 THEN 1 WHEN TRANSACTIONTYPE = 32 THEN -1 ELSE 0 END AS TRANSFER, CASE WHEN TRANSACTIONTYPE = 30 THEN 1 ELSE 0 END AS WITHDRAW, CASE WHEN TRANSACTIONTYPE = 33 THEN 1 ELSE 0 END AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

LEFT JOIN TRANSACTIONS TRANSACTIONS ON TRANSACTIONS.Membership\_ID = CUSTOMER\_MEMBERSHIP\_DATES.Membership\_ID

AND TRANSACTIONS.Package\_ID = CUSTOMER\_MEMBERSHIP\_DATES.Package\_ID

WHERE CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE >={ts '2014-07-01 00:00:00'}

AND CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE <={ts '2014-07-31 00:00:00'}))

ORDER BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender

## SQL of Count entire membership as one single unit (1)

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

COUNT(\*) AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMER\_MEMBERSHIP\_DATES.CUSTOMER\_ID

WHERE CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE <= {ts '2014-07-31 00:00:00'}

GROUP BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender)

UNION

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

COUNT(\*) AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

WHERE CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE <{ts '2014-07-01 00:00:00'}

GROUP BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender)

UNION

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

COUNT(\*) AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

WHERE EFFECTIVE\_DATE <{ts '2013-07-31 00:00:00'}

GROUP BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender)

UNION ALL

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1,

SITES.sitename AS group2,

PACKAGES.packagename AS group3,

CUSTOMERS.gender AS group4,

0 AS NEW,

0 AS TRANSFER,

0 AS WITHDRAW,

0 AS RENEWAL,

CASE

WHEN EXPIRATION\_DATE >={ts '2014-07-01 00:00:00'}

AND EXPIRATION\_DATE <={ts '2014-07-31 00:00:00'} THEN 1

ELSE 0

END AS EXPIRE,

CASE

WHEN DATESUSPENDEDFROM >={ts '2014-07-01 00:00:00'}

AND DATESUSPENDEDFROM <={ts '2014-07-31 00:00:00'} THEN 1

ELSE 0

END AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID)

UNION ALL (

(SELECT PACKAGE\_CATEGORIES.categoryname AS group1, SITES.sitename AS group2, PACKAGES.packagename AS group3, CUSTOMERS.gender AS group4, CASE WHEN TRANSACTIONTYPE = 29 THEN 1 ELSE 0 END AS NEW, CASE WHEN TRANSACTIONTYPE = 31 THEN 1 WHEN TRANSACTIONTYPE = 32 THEN -1 ELSE 0 END AS TRANSFER, CASE WHEN TRANSACTIONTYPE = 30 THEN 1 ELSE 0 END AS WITHDRAW, CASE WHEN TRANSACTIONTYPE = 33 THEN 1 ELSE 0 END AS RENEWAL,

0 AS EXPIRE,

0 AS SUSPEND,

0 AS TOTALMEMBERCURRENT,

0 AS TOTALMEMBERBEFORE,

0 AS TOTALMEMBERLASTYEAR

FROM PACKAGES "PACKAGES"

LEFT JOIN PACKAGE\_CATEGORIES PACKAGE\_CATEGORIES ON PACKAGE\_CATEGORIES.PackageCategory\_ID=PACKAGES.PackageCategory\_ID

LEFT JOIN SITES SITES ON SITES.SITE\_ID=PACKAGES.SITE\_ID

RIGHT JOIN CUSTOMER\_MEMBERSHIP\_DATES CUSTOMER\_MEMBERSHIP\_DATES ON CUSTOMER\_MEMBERSHIP\_DATES.PACKAGE\_ID=PACKAGES.PACKAGE\_ID

JOIN CUSTOMERS CUSTOMERS ON CUSTOMERS.Customer\_ID=CUSTOMER\_MEMBERSHIP\_DATES.Customer\_ID

LEFT JOIN AGE\_CATEGORIES AGE\_CATEGORIES ON CUSTOMERS.AGE\_CATEGORY\_ID=AGE\_CATEGORIES.AGE\_CATEGORY\_ID

JOIN MEMBERSHIPS MEMBERSHIPS ON MEMBERSHIPS.PRIMARYMEMBERCUSTOMER\_ID = CUSTOMERS.CUSTOMER\_ID

AND MEMBERSHIPS.MEMBERSHIP\_ID = CUSTOMER\_MEMBERSHIP\_DATES.MEMBERSHIP\_ID

LEFT JOIN TRANSACTIONS TRANSACTIONS ON TRANSACTIONS.Membership\_ID = CUSTOMER\_MEMBERSHIP\_DATES.Membership\_ID

AND TRANSACTIONS.Package\_ID = CUSTOMER\_MEMBERSHIP\_DATES.Package\_ID

WHERE CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE >={ts '2014-07-01 00:00:00'}

AND CUSTOMER\_MEMBERSHIP\_DATES.EFFECTIVE\_DATE <={ts '2014-07-31 00:00:00'}))

ORDER BY PACKAGE\_CATEGORIES.categoryname,

SITES.sitename,

PACKAGES.packagename,

CUSTOMERS.gender